International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 4, Issue 2, Apr 2014, 133-142

TRANS STELLAR
• Journal Publications • Research Consultancy

IDENTITY-BASED CRYPTOGRAPHY FOR SECURE MOBILE COMMUNICATION

RUSHIKESH MADHUKAR BAGE & GAURAV DIGAMBAR DOIPHODE

Graduate Student, Department of Computer Science, Mumbai, Maharashtra, India

ABSTRACT

In this paper, an identity-based key agreement system and its implementation for mobile telephony in GSM and UMTS networks is presented. The use of telephone numbers as public keys allows the system to piggyback much of the security overhead for key management to the existing GSM or UMTS infrastructure. The proposed approach offers solutions to the problems of multi-domain key generation, key distribution, multi-domain public parameter distribution and inter-domain key agreement. The feasibility of the approach is illustrated by presenting experimental results based on smartphones. While it is possible to implement end-to-end encryption of mobile phone calls based on a Public Key Infrastructure (PKI), the complexity of setting up and using a PKI is prohibitive, especially since many users of mobile phones are not well versed in cryptographic procedures and are quickly overwhelmed when confronted with public and private keys, certificates, signatures and revocation lists.

KEYWORDS: Cryptography, Security, Mobile, Calls, GSM, UMTS, Identiy-Based, Smartphones

1. INTRODUCTION

In mobile phone networks, eavesdropping on a call is easy, even for non-governmental forces. Since the encryption schemes in GSM (2G) and UMTS (3G) only encrypt calls between the mobile phone and the base station, an attacker positioned any-where in the network between the two base stations can usually intercept calls without great difficulty. Furthermore, since GSM base stations are not authenticated, an attacker can pose as a base station and intercept phone calls in the vicinity. Due to backwards compatibility and UMTS coverage issues, most UMTS devices allow network fallback to GSM, opening up UMTS devices to the same man-in- the-middle attacks that afflict GSM networks.

Identity-based cryptography (IBC) promises to offer an approach to end-to-end encryption for mobile telephone calls in which the telephone numbers of the call participants are used as the public keys to secure the communication channel, thus making the cryptographic security procedure as easy as making a telephone call. The use of telephone numbers as public keys has two major benefits. Firstly, since the caller knows the number to be called, the caller also automatically knows the public key and does not need a separate public key look-up or certification infrastructure. Secondly, telephone numbers are easy to understand and users are confident in using them, such that there is no need to educate users to understand the link between a telephone number, a public key and/or its certificate, thus significantly lowering the complexity threshold of phone call encryption.

Several solutions have been proposed which allow multiple identity private key generator (ID-PKGs) to interoperate [1–3], but these systems require either co-operation between the ID-PKGs or a hierarchical approach with a trusted party at the top. Both of these approaches are difficult to use in the area of mobile telephony due to organizational difficulties and conflicting business interests. As demonstrated by approaches based on a Certificate Authority (CA), there will always be competing organizations offering the same service for the same protocol (e.g. signing RSA public keys)

without wanting to cooperate on the corporate level. Thus, to successfully deploy IBC in mobile telephony, the IBC system must be able to cope with the real world network issues, such as allow competing organizations to operate their ID-PKG independently of other ID-PKGs, roaming and changing providers while still enabling cross-domain execution of the IBC protocols for their customers.

In this paper, a new multi-domain identity-based key agreement system is introduced which focuses on the issues to be solved when implementing IBC for mobile telephony. The proposed approach is realized using standard telephone numbers as public keys with multiple security domains (i.e. mobile telephony providers). It utilizes the mathematics also used in the traditional Diffie-Hellman key agreement and Rivest-Shamir-Adleman (RSA) public key cryptography approaches. Solutions to the problems of multi-domain key generation, key distribution, multi-domain public parameter distribution and inter-domain key agreement are presented.

2. PROBLEM STATEMENT

In GSM networks, communication between a mobile system (MS) (i.e. a mobile phone) and a base transceiver station (BTS) is encrypted using the A5 cryptographic protocol. Due to design flaws, A5 is vulnerable to crypto analysis such that hackers can eavesdrop on the communication. Updates to the A5 protocol have been proposed to hinder further attacks, and the UMTS standard has replaced A5 by a more secure (and open) protocol, making cryptographic attacks less of a concern.

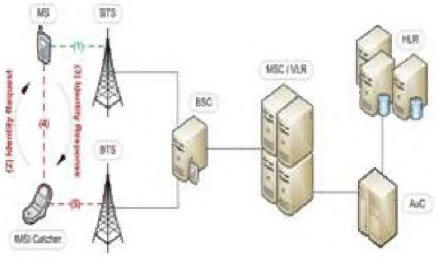

Figure 1: IMSI Catcher Attack

A simpler attack is to subvert the communication setup before encryption. To allow a MS to authenticate itself to the network provider, it gets a subscriber au- thentication key (SAK). The SAK is stored both on the SIM card of the MS and in the Home Location Register (HLR) of the provider. The BTS are connected to a Base Station Controller (BSC) that in turn is connected to a Mobile Switching Center (MSC) and a Visitor Location Register (VLR). These in turn are connected to the HLR and the Authentication Center (AuC) that give access to the SAK of the MS. During the authentication process, a 128-bit random number is generated which using the A3 [7] is combined with the SAK to create a 32-bit authentication key called SRES. The SRES key is then sent to the BTS. The SRES key is then compared to the SRES* key that is computed by the AuC of the provider also using the A3 algorithm and the HLR SAK. If the two values match, the MS is authenticated and may join the network. The BTS does not authenticate itself to the MS. This opens up the possibility of a Man-in-the-Middle (MITMA) attack. Using an IMSI catcher [8], an attacker can pose as a BTS and intercept calls in the vicinity by broadcasting a strong base station signal. Figure 1 shows the procedure. MS are programmed to connect to the strongest BTS signal, thus if the IMSI catcher has the strongest signal they serve their current BTS connection (1) and will connect to the IMSI catcher (2) no questions asked (3). Since the BTS is also responsible for selecting the security mechanism, the IMSI catcher can then force the MS to turn off or select an insecure encryption algorithm (4) and thus allow the MITMA to operate.

The downside to this attack is that the IMSI catcher cannot function as a real BTS since it is not connected to the main phone network and must forward calls using its own MS and SIM (5). However, since the SIM in the IMSI catcher cannot register itself as the target SIM (due to the authentication of the MS), the attacked MS is not registered at any BTS and is not reachable while it is connected to the IMSI catcher. Thus, only outgoing calls can be intercepted, since the network cannot reach the attacked MS. Furthermore, the IMSI catcher is not a targeted attack. It affects all MS in its vicinity all of which are not reachable while they are connected to the IMSI catcher and whose calls would need to be forwarded if the IMSI catcher is not to become noticeable. While this attack should not be taken lightly, there are some real world problems in its execution.

A much simpler attack is enabled by cost saving measures in common practice when setting up base stations. Since connecting all BTS to a secured wired network is costly, BTS can also be connected to the main network via a directed microwave link. This microwave signal is sent without encryption and can easily be intercepted, giving an attacker clear text access to all calls going via this link without leaving a physical trace. But even a wired connection is not safe if an attacker is willing to apply a physical tap to the line. These link taps are particularly relevant since they can be used without affecting the rest of the network and thus cannot be easily detected. They also allow a large number of calls to be tapped simultaneously. For instance, a BTS located near a firm, government building or celebrity house can be tapped, thus, making all mobile calls made to and from that location available to the attacker. Since the equipment needed to execute such a tap is be- coming more portable and cheaper at a rapid rate, this kind of attack will rapidly gain in relevance.

To prevent the above attacks, end-to-end protection of phone calls is required. However, the solution must be able to be deployed in a multi-organization environment and be usable by non-tech savvy users. As stated in the introduction, conventional PKI based solutions are too complex both for the network providers and for the users. A simple approach is required which can be implemented by net- work providers independently of each other and which does not introduce added complexity for end users. In the next section, an algorithm will be presented that fulfills these requirements. The algorithm allows two MS to perform a session key- agreement over an unsecured channel and between different providers using telephone numbers as public keys. Using the created session key, a symmetric encryption of all call data can be performed. The algorithm prevents MITMA attacks and offers perfect forward security.

3. ALGORITHMS

3.1 Algorithmic Overview

The identity-based key agreement protocol SSF (Secure Session Framework) consists of four main algorithms: Setup, Extract, Build SIK, and Compute.

3.2 Key Agreement

The Setup algorithm (Figure 2) is executed by the ID-PKG. This part of the key agreement protocol is only performed once and creates both the master secrets P and Q as well as the public parameters.

Setup - The Setup algorithm is executed by the ID-PKG.

Input: k ? N

Step 1: Choose an arbitrary integer R > 1 from Z+ .

Step 2: Generate two primes, P and Q, of bit length k with the following properties:

1. The prime factorization of (P - 1) contains a large prime P

2. The prime factorization of (Q - 1) contains a large prime Q

3. gcd R = 1, where φ is the Totient Function.

Step 3: Compute the product N = PQ

Step 4: Choose a generator G of a subgroup G of ZN whose order contains at least one of

the primes P or Q such that the Computational Diffie Hellman Assumption (CDHA) [9]

holds in G.

Step 5: Choose a cryptographic collision-resistant hash function H : {0, 1}* ? ZN .

Output: PSP = (N, G, R, H(·)), SP = {P, Q}

Figure 2: Setup Algorithm

Public, Shared Parameters. The public, shared parameters (PSP) of a domain D of the key agreement protocol SSF is the quadruple PSP = (N, G, R, H(·)).

The Extract algorithm (Figure 3) creates the identity key (i.e. the private key) for a given identity. This algorithm is executed by the ID-PKG. If all IDs are known and the range is not too big (e.g. a Class B or C subnet of the Internet), it is possible to execute this step for all IDs offline, and the master secrets can then be destroyed, if required.

Extract - The Extract algorithm is executed by the ID-PKG.

Input: PSP, SP, ID

Let ID be a given identity. The algorithm computes dID = H(ID)1/R

dID is called the identity key and is given to the entity EID .

Ouput: dID

Figure 3: Extract Algorithm

The Build SIK algorithm (Figure 4) is executed by the devices taking part in the key agreement.

The random integer rID is generated with a secure number generator to make rID unpredictable. The private

identity key is used in combination with this randomly chosen integer and the generator in such a way that it is not possible to extract the identity key from the SIK. This is due to the fact that the multiplications are performed in the ring ZN and the result set of a division in the ring ZN is so large that the extraction of the identity key is infeasible. The SIK is then sent over an unsecured channel to the other party and vice versa. The SIK must be greater than zero to prevent a trivial replacement attack where an attacker replaces the SIKs with zero which in turn would make the session key zero as well. Any other replacement attacks lead to invalid session keys. The final step of the key agreement process is the computation of the session key using the Compute algorithm (Figure 5) which is executed by the devices taking part in the key agreement. By applying the inverse of the hash value of the opposite's identity, the involved identity key is canceled out. Only if both endpoint addresses match their identity keys, a valid session key is created.

Build SIK - The Build SIK algorithm is executed by the entity EID

Input: PSP, dID

Step 1: Choose a random integer rID from Z+ .

Step 2: Compute SIKID = GrID · dID (mod N).

SIKID is the SIK (session initiation key) for the identity string ID that belongs to entity EID .

Output: SIKID

Figure 4: Build SIK Algorithm

Compute - The Compute algorithm is executed when two parties are performing a key agreement

Input for EID1 : EID2 , PSP, SIKID2 , rID1

Input for EID2 : EID1 , PSP, SIKID1 , rID2

When EID1 receives the session initiation key from EID2 , it calculates

(SIK2 · H(ID2)- 1)rID1 = (((GrID2 · dID2)R · H(ID2)- 1)rID1 = GRrID1 rID2 = S mod N

When EID2 receives the session initiation key from EID1 , it calculates

(SIK1 · H(ID1)- 1)rID2 = (((GrID1 · dID1)R · H(ID1)- 1)rID2 = GRrID1 rID2 = S mod N

Output: H(S), the common session key for EID1 and EID2 .

Figure 5: Compute Algorithm

The key distribution system proposed by Okamoto [10] extracts its identity information in a similar manner as in our scheme, but Figure does not address the case of key agreement between different domains.

3.3 Key Agreement between Different Domains

The ID-PKG determines the public, shared parameters, and all entities that receive their identity key for their IDs from this generator can establish a key agreement among each other. In practice, it is very unlikely that all phones will receive their identity key from the same security domain, since this would imply the existence of a third party trusted by all with a secure communication link to all devices. Since telephone network providers are in charge of managing the MS

information of their customers autonomously, it is desirable that they also manage the security information autonomously, meaning that they must be allowed to operate their own ID-PKG without having to cooperate with other providers. The management infrastructure, such as HLRs and AuC, can then simply be extended by the required additional data.

We now show how cross-domain key agreement can be achieved such that only the public parameters must be distributed (which will be discussed in section 4). Each device only needs a single identity key, and the ID-PKGs do not need to agree on common parameters or participate in any form of hierarchy. In the following, we assume without loss of generality, that there are two domains D1 and D2.

Their public parameters are (N1, G1, R1, H1 (·)) and (N2, G2, R2, H2 (·)), respectively. Every parameter can be chosen independently. The case that (R2, φ (N1)) > 1 or (R1, φ (N2)) > 1 is not critical, since no Rth roots must be computed regarding the other domain's modulus. The two moduli N1 and N2 were chosen according to the requirements stated in the Setup algorithm, i.e. the computation of discrete logarithms is infeasible in ZN1 and ZN2, respectively. Consequently, an algorithm such as the Pohlig Hellman algorithm [11] cannot be applied and Pollard's P − 1 factoring algorithm [12] will not be a threat. Thus, a random non-trivial integer has a large order in ZN1 N2 with an overwhelming probability, and the computation of discrete logarithms is infeasible in ZN1 N2.

In the following, an entity EID1 from D1 wants to communicate with EID2 from D2. The algorithm for cross-domain key extension is shown in Figure 6

Cross-Domain Key Extension (from the view of participant EID1)

Executes: Query rPSP, Extend IK and Build eSIK

Input: PSP1 , PSP2 , dID1

Step 1: Calculate the common, shared, public parameters: PSP1,2 = (N1 · N2 , G1 · G2 , R1 ·R2 , H2 (·)).

Step 2: Use the Chinese-Remainder Theorem to calculate the integer dID1 :

dID1 = dID1 mod N1 and dID1 = 1 mod N2

Step 3: Use the Chinese-Remainder Theorem to calculate the integer H1 (ID):

H1 (ID1) = H1 (ID1)R2 mod N1 and H1 (ID1) = 1 mod N2 (1,2)

Step 4: Build eSIK via eSIKID1 = (G1 · G2)rID1 dID1 mod N1 N2

Output:(1,2)eSIKID1 ,the cross-domain session initiation key.

Figure 6: Cross-Domain Key Extension Algorithm

In step 1 of the cross-domain key agreement algorithm, the common shared public parameters are the element-wise product of both sets of domain parameters. In step 2, entity EID1 extends its identity key using the Chinese-Remainder Theorem. In step 3, entity EID1 extends its hash identifier also using the Chinese- Remainder Theorem. The procedure for entity EID2 is analog, only the indices change from 1 to 2. Key agreement is then performed using the extension of the original algorithm shown in Figure 6.

4. IMPLEMENTATION ISSUES

Like most other IBC approaches, our system also uses shared public parameters. In a single domain scenario, the

distribution of the public parameters is not a problem. However, if each network provider runs its own ID-PKG, the number of public parameters and the binding between public parameters and identity keys becomes more complex. As stated above, this distribution problem is still much smaller than the distribution problem for traditional public keys where each entity has its own public key that needs to be distributed. Of course, traditional PKI technology can be used to distribute the public parameters, but a more suitable solution is to integrate the public parameters into the GSM/UMTS lookup mechanism and carry the information over the SS7 protocol.

Since there already is lookup functionality to locate the HLR of a MS and the current location of the MS, a flag can be attached to the request message, stating that the public parameters of the MS should be sent piggybacked to the response. The flag is used, since the public parameters only need to be queried for the very first call to a MS of a particular provider. All subsequent calls to the same or other MS of the same provider do not need a further public parameter lookup. In the case of UMTS, this is reasonably secure since the BTS must authenticate itself to the MS and thus an active MITMA is prevented that could otherwise tamper with the public parameters.

The passive MITM As still possible with UMTS are not a danger to the transfer of the public parameters since they are public anyway. In the case of GSM, this form of public parameter distribution holds the risk of an attacker with an IMSI catcher replacing the public parameters with his own on the first call made to a provider by a MS. However, this attack only works on the very first call ever placed to a provider and will be detected as soon as the MS calls someone else at the same provider after the attack due to a public parameter mismatch. To summarize, this form of public parameter distribution is not a problem in UMTS networks and if the slight security risk in GSM networks is unacceptable, a traditional CA based signing approach can be added to prevent tampering with the public parameters.

4.1 Key Expiration

Another practical issue of mobile phone call encryption is the fact that telephone numbers are reused. In a PKI or CA based solution, this creates several problems, since the central PKI must be updated or the CA must be contacted to resign public keys as the MS swap telephone numbers. Certificate Revocation Lists can be used to accomplish this, however the solutions tend to become quite complex. In particular, public key caching mechanisms can lead to problems.

In the presented identity-based solution, natural key expiration techniques can be used to cope with telephone number reuse. Boneh et al. showed how keys can be given a lifetime, which allows natural expiration of the identity key. This is done by the internal concatenation of the ID, in our case the telephone number, with a date. The same technique can be used in our solution. Thus, when a customer releases a telephone number and it is reused, the next customer will have a different identity key based on the current date. Since telephone number reuse is time-delayed in any case, this time frame can be used as the key lifetime to ensure that each successive owner lies in a new lifetime slot. With the techniques introduced in this paper, a frequent automatic in-band key distribution can be safely executed and thus key renewal is far less of a problem. Additionally, key expiration also reduces the risk of identity key theft, since the attack window is restricted to a small time interval.

5. EXPERIMENTAL RESULTS

In this section, experimental results of the presented identity-based cryptographic security solution for mobile phone key agreement are presented. The experiments for the key agreement, the following parameters were examined: the modulus - with N = 512, 1024, 2048 and 4096 Bit, the random exponent - with rID = 64, 128, 256 and 512 Bit and the

chosen public parameter R = {3, 17, 513, 65537}. The numbers chosen for R were selected to give an overview of the performance of the algorithm based on the size of R. R can be chosen arbitrarily by the ID-PKG according to the setup algorithm (Step 2.3). Each of the following tables contains the mean time for the key agreement operations of the 100 trial runs computed using a fixed modulus with rID and R in the rows and columns. It is evident from the tables that the main contribution to the computational time is the modulus and the random exponent.

The public random number R selected by the provider does not have a significant effect due to the fact that the computational time of the algorithm depends on the number of 1 s in the binary representation of the number and the used random numbers all contain two bi- nary 1 s. The random number R is not security critical for R > 3. While the time needed for key agreement using a 4096-bit modulus and a 512-bit random exponent is too long for current devices, key agreement with a 2048-bit modulus and 128 or 256-bit random exponents has acceptable run times. Once a session key has been established, a symmetric encryption of the call using AES 256 is executed. The encoding block was set to 4096 Byte which contains at least 256 ms (depending on the compression) of audio data.

6. RELATED WORK

Kumar et al. present an IBC based approach to mutual authentication and key agreement for GSM networks. Unlike our proposal, Kumar et al. use the IMSI number as the public identity key. The security of the protocol relies on a secure channel to the HLR and VLR (Phase 1, Steps 2 and 3). Both these design decision have drawbacks. Firstly, using the IMSI as the public key means the users must trust the infrastructure to show them the correct binding between telephone number and IMSI number, since most users do not know their own IMSI, let alone the IMSI of other users. Secondly, the communication channels between the MS and the HLR and VLR are not considered to be secure and must be handled by the presented solution.

There are other approaches such as the Cryptophone that applies the Zfone VoIP security mechanism to mobile phones. ZFone executes a standard Diffie Helmann key agreement (which is vulnerable to an active MITMA), but then dis- plays a hash of the generated session key to both users. One user must then read out the hash to the other user, who can then see if the key agreement was com- promised, since if a MITMA attack has taken place, the hash values are different. While preventing simple MITMAs, the ZFone solution is somewhat cumbersome, since users must read out hash values to each other. It also does not prevent impersonation attacks or voice based MITMA attacks. The key distribution system proposed by Okamoto extracts its identity in- formation in a similar manner as in our scheme, but does not address the case of key agreement between different domains.

7. CONCLUSIONS

In this paper, an identity-based key agreement system for mobile telephony in GSM and UMTS networks was presented. All attacks presented in the paper can be successfully prevented by the identity-based cryptographic solution. The use of telephone numbers as public keys reduced the complexity of the security management framework and well as the usage complexity for phone call encryption.

The approach offers solutions to the real world problems in realizing an identity-based security framework for mobile phone call encryption, namely multi-domain key generation, key distribution, multi-domain public parameter distribution and inter-domain key agreement. Experimental results showing that

Current smartphones are powerful enough to run the presented system. Future work will include simulated large scale deployment and scalability studies to quantitatively evaluate the administrative benefit of using the presented identity-based approach compared to a traditional PKI. The proof-of-concept solution will also be ported to further platforms beyond Symbian. Finally, user-studies will be performed to further evaluate the benefits to the non-tech savvy end user.

REFERENCES

1. J. Horwitz and B. Lynn, "Toward Hierarchical Identity-Based Encryption," in EURO-CRYPT '02: Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques. Springer-Verlag, 2002, pp. 466–481.
2. N. McCullagh and P. Barreto, "A New Two-Party Identity-Based Authenticated Key Agreement," in Cryptographers' Track at RSA Conference - CT-RSA, 2005.
3. D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext ," in Advances in Cryptology - Eurocrypt 2005, Lecture Notes in Computer Science, vol. 3494. Springer-Verlag, 2005, pp. 440–456.
4. W. Diffie and M. E. Hellman, "New Directions In Cryptography," IEEE Transactions On Information Theory, no. 6, pp. 644–654, 1976.
5. R. L. Rivest, A. Shamir, and L. Adleman, "A Method For Obtaining Digital Signatures And Public Key Cryptosystems," Communications Of ACM, vol. 1, no. 2, pp. 120–126, 1978.
6. S. Petrovic, "An improved Cryptanalysis of the A5/2 Algorithm for Mobile Communications," in Proceedings of the IASTED International Conference on Communication Systems and Networks, 2002, pp. 437–444.
7. C. Clavier, "An Improved SCARE Cryptanalysis against a Secret A3/A8 GSM Algorithm," in Third International Conference on Information Systems Security, 2007, pp. 143–155.
8. U. Meyer and S. Wetzel, "A Man-In-The-Middle Attack on UMTS," in WiSe '04: Proceedings of the 3rd ACM Workshop on Wireless Security. New York, NY, USA: ACM, 2004, pp.90–97.
9. F. Bao, R. H. Deng, and H. Zhu, "Variations of Diffie Hellman Problem," in International Conference on Information and Communications Security, 2003, pp. 301–312.
10. E. Okamoto, "Key Distribution Systems Based on Identification Information," in CRYPTO '87: A Conference on the Theory and Applications of Cryptographic Techniques on
11. Advances in Cryptology. London, UK: Springer-Verlag, 1988, pp. 194–202.
12. S. Pohlig and M. Hellman, "An Improved Algorithm for Computing Logarithms over
13. GF (p) and its Cryptographic Significance," 1984, pp. 106–110.
14. J. Pollard, "Theorems of Factorization and Primality Testing," Mathematical Proceedings of the Cambridge Philosophical Society, vol. 76, pp. 521–528, 1974